Python Literals –

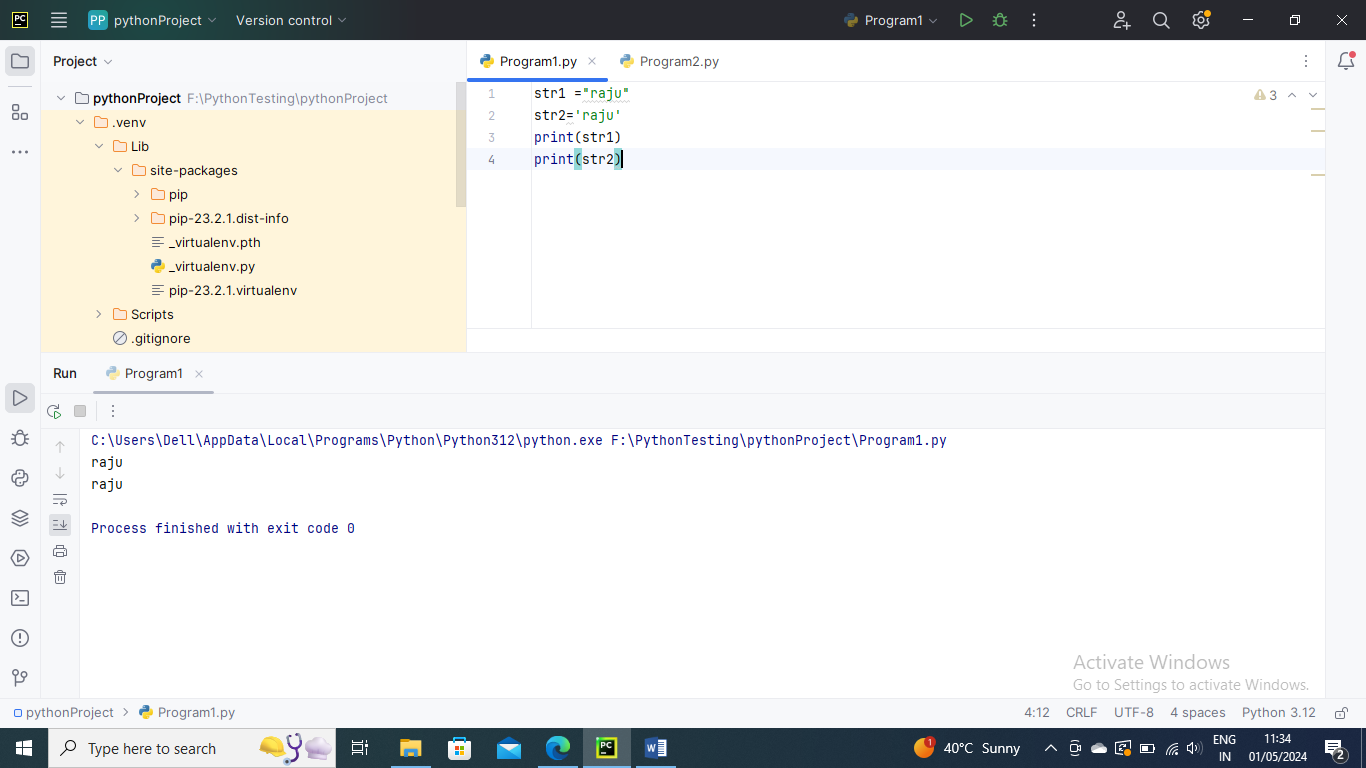
Python Literals can be defined as data that is fiven in a variable or constant

Python supports the following Literals

1. String Literals

String literals can be formed by enclosing a text in doend uble quotes. We can use both single quotes as well as double quotes to create a string

“raju” , ‘raju’



Type of Strings –

a) Single-line String – Strings that are terminated with in a single line

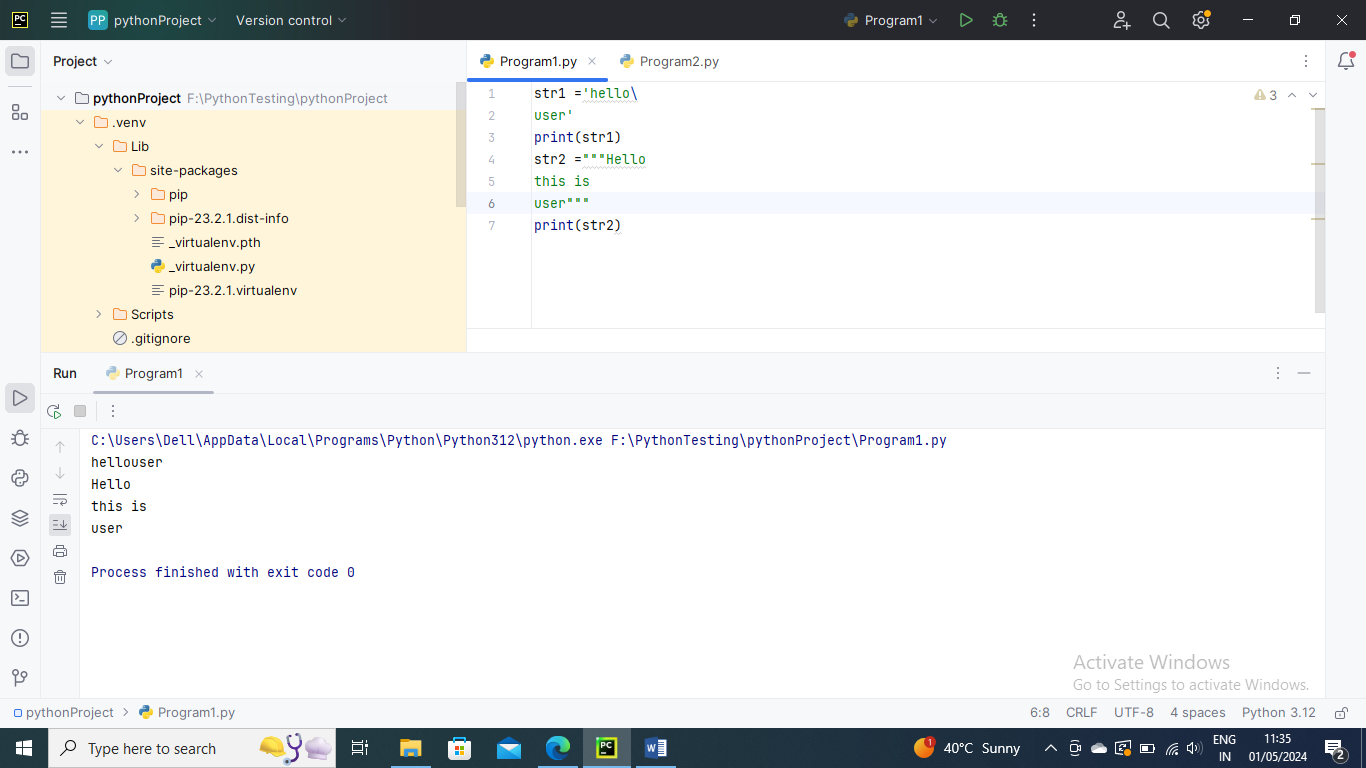
text1 =’hello’

b) Multi-line String – A piece of text written in multiple lines

two –ways

1) Adding black slash at end of each line

2) using triple quotation marks



Numeric Literals –

Int – Numbers can be positive and negative

ex: 100

Long – Integers of unlimited size followed by lower or uppercase L

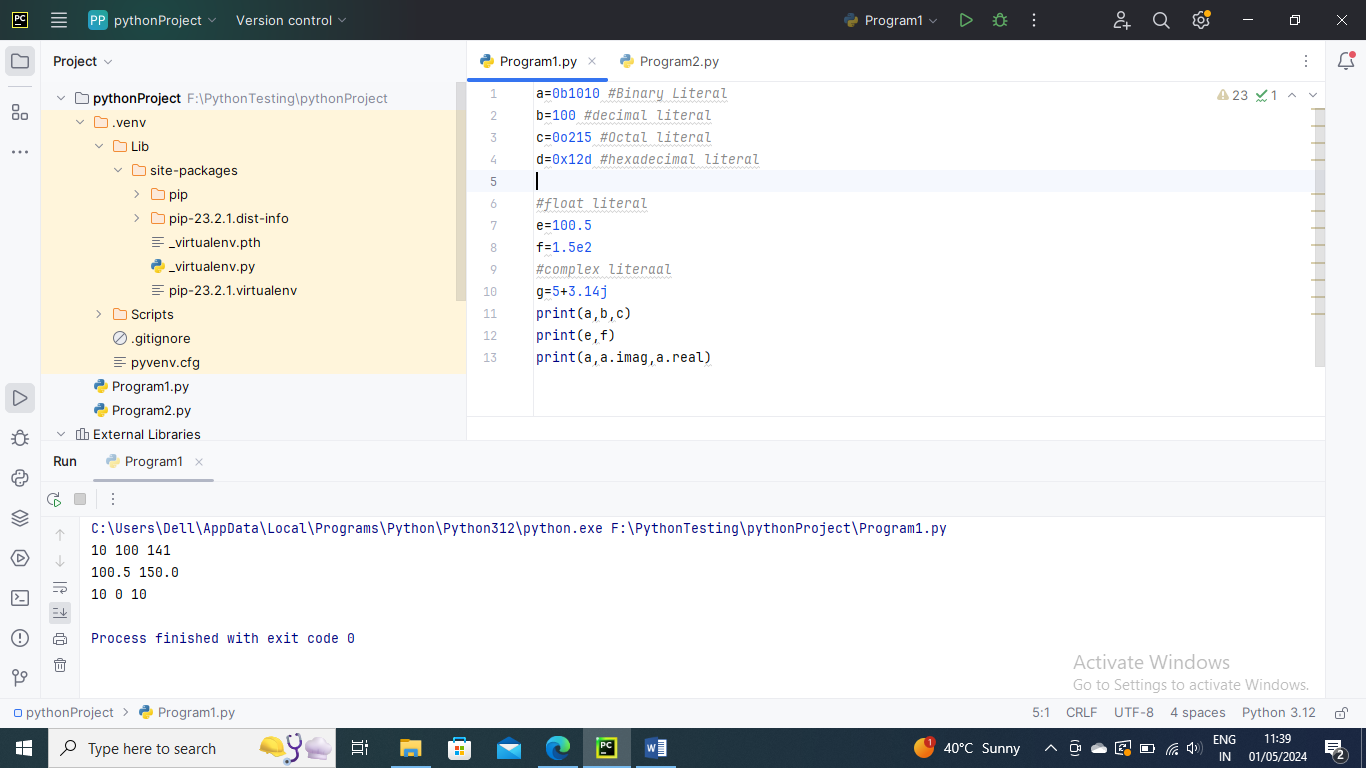
Ex: 8702365L

Float – real numbers with integer and fractional part

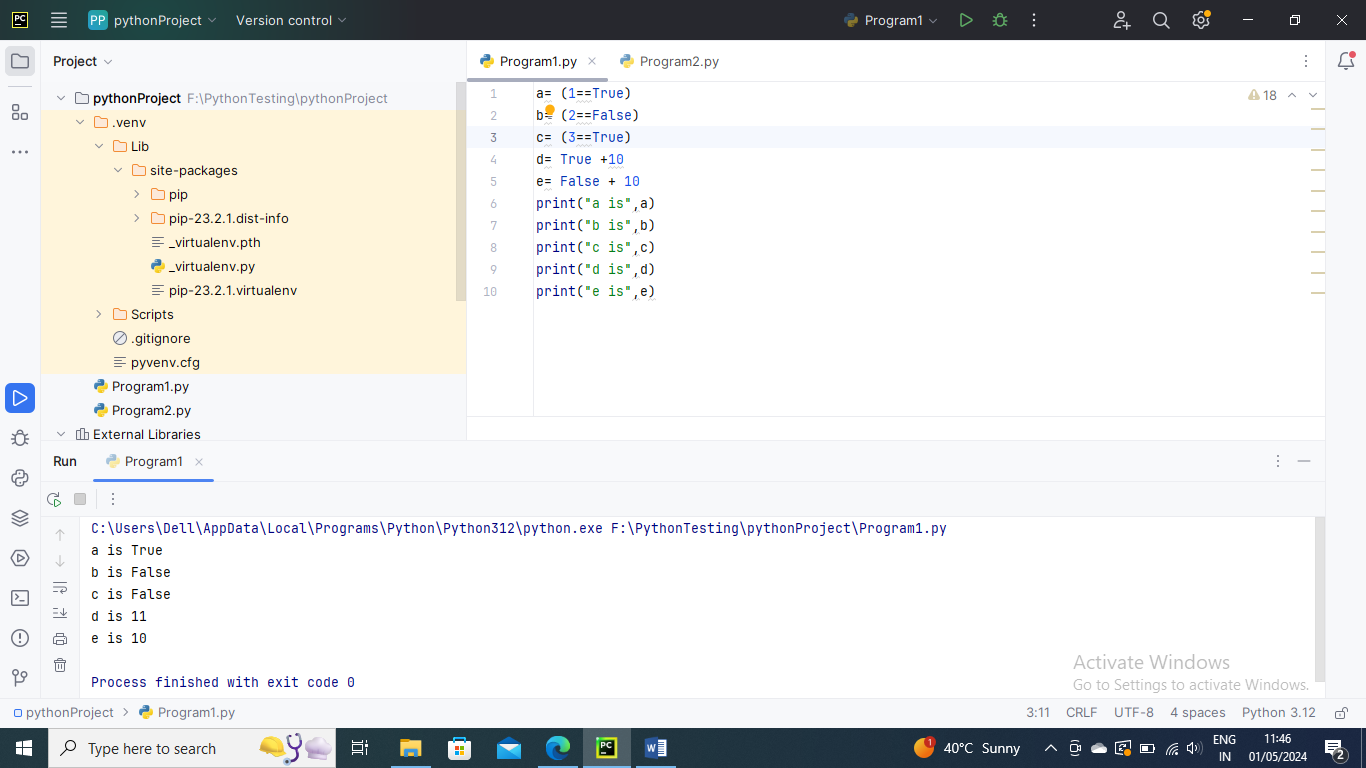
Ex: -26.2

Complex- In the form of a+bj where a forms real part and b forms the imaginary part of the complex number

Ex: 3.14j



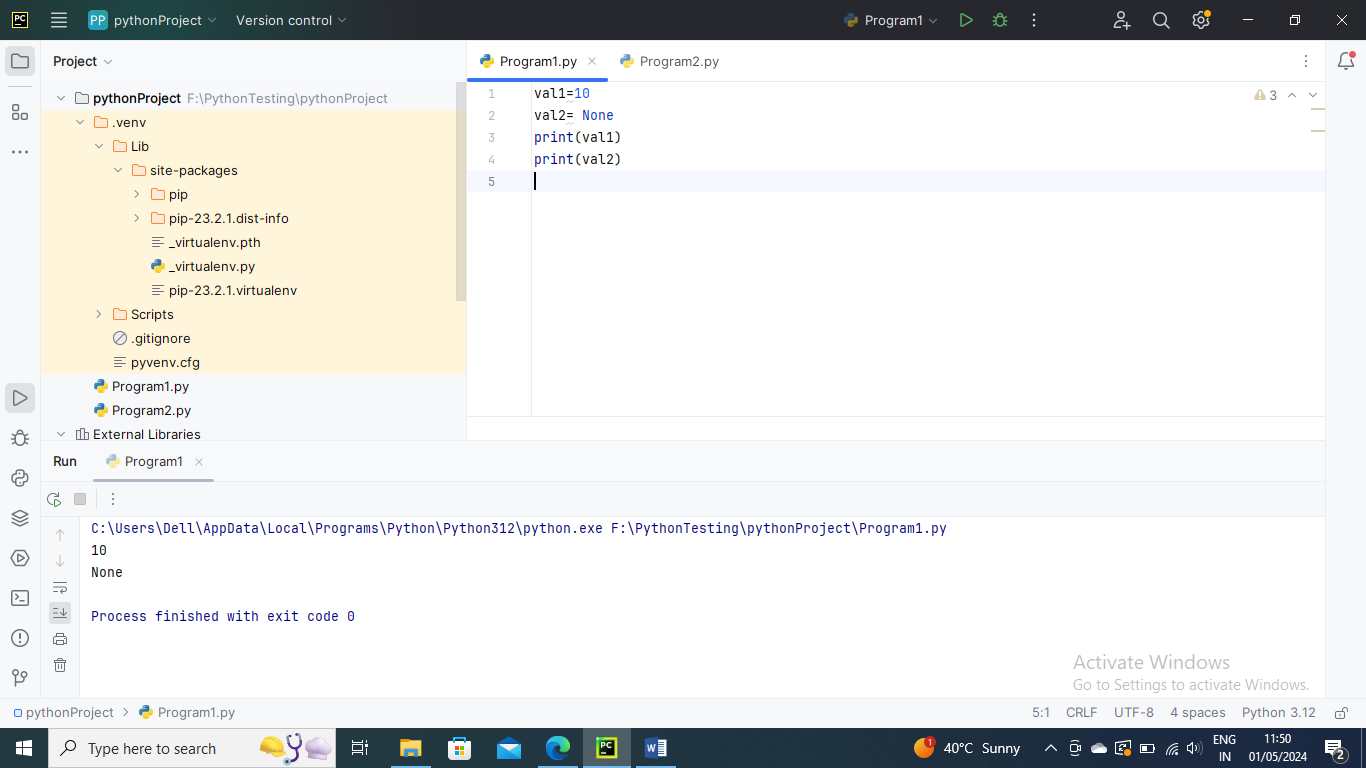
Boolean Literals-



Special Literals –

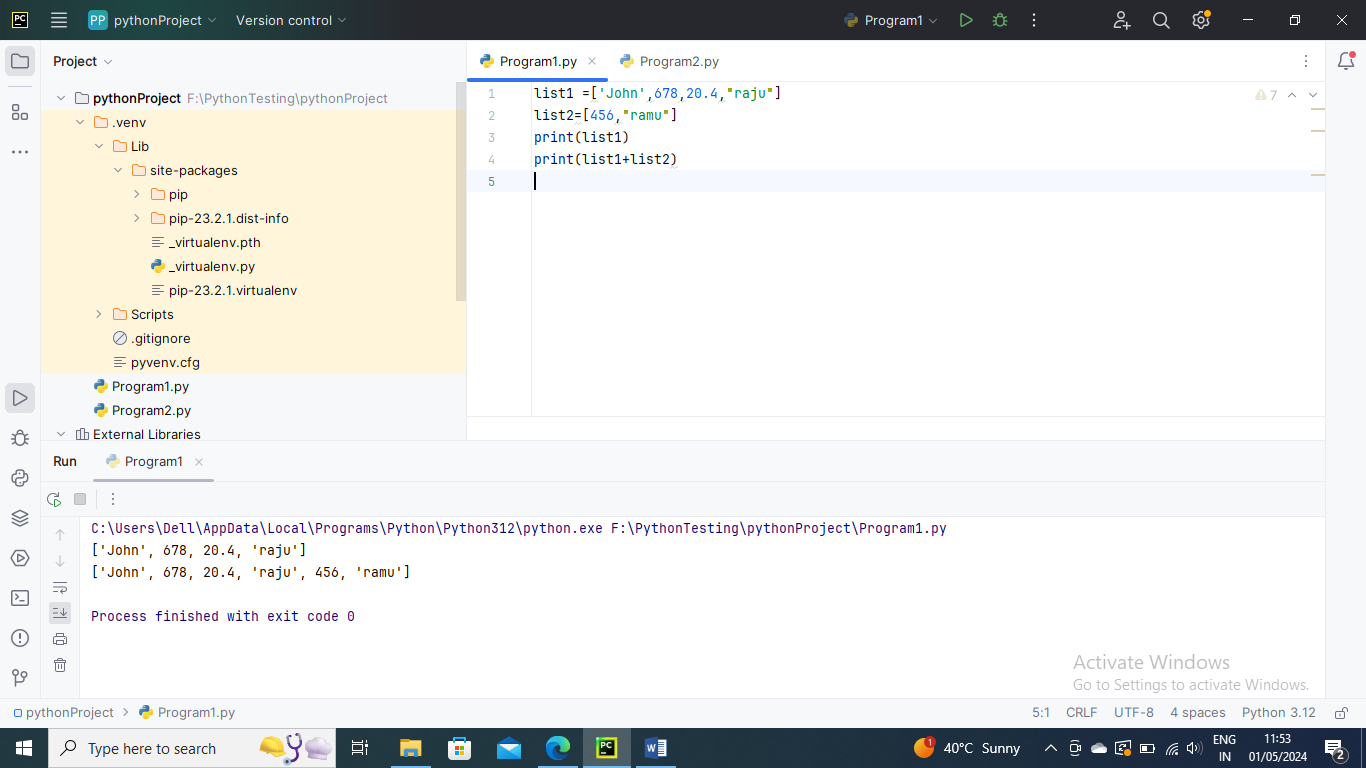
Python contains one special literal i.e. None

None is used to specify to that field is not created. It is alose used for the end of lists in Python

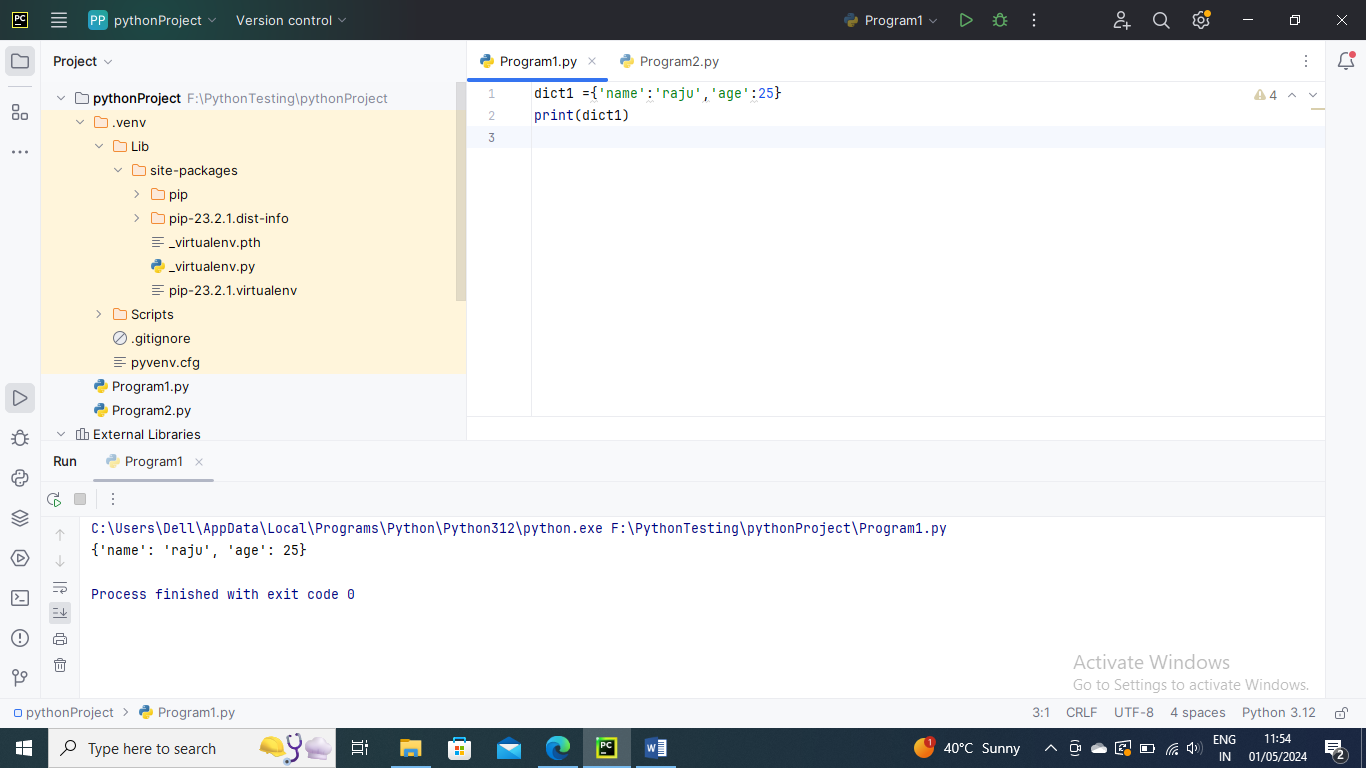


Literal Collections

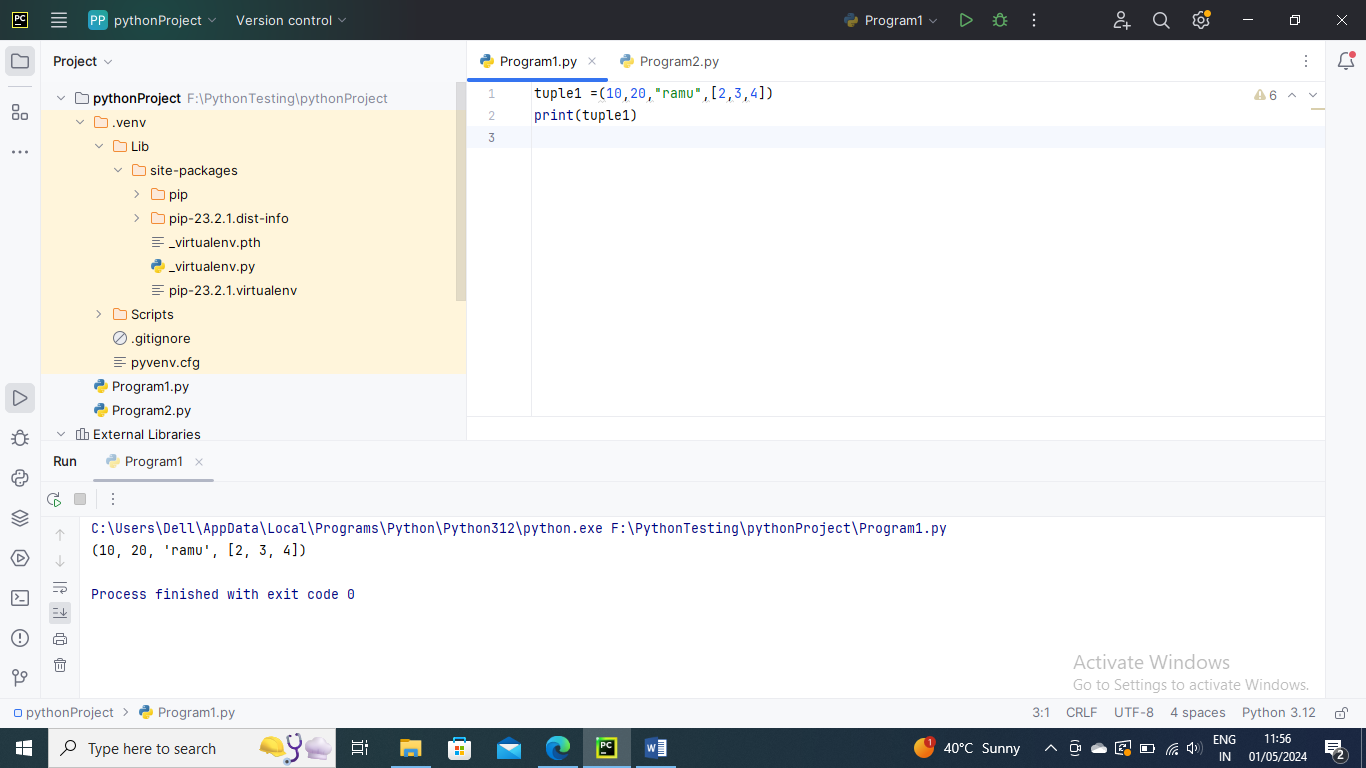
Python supports four types if Literal collection such as List literals, tuple Literals, Dict Literals and Set Literals

List Literal – 

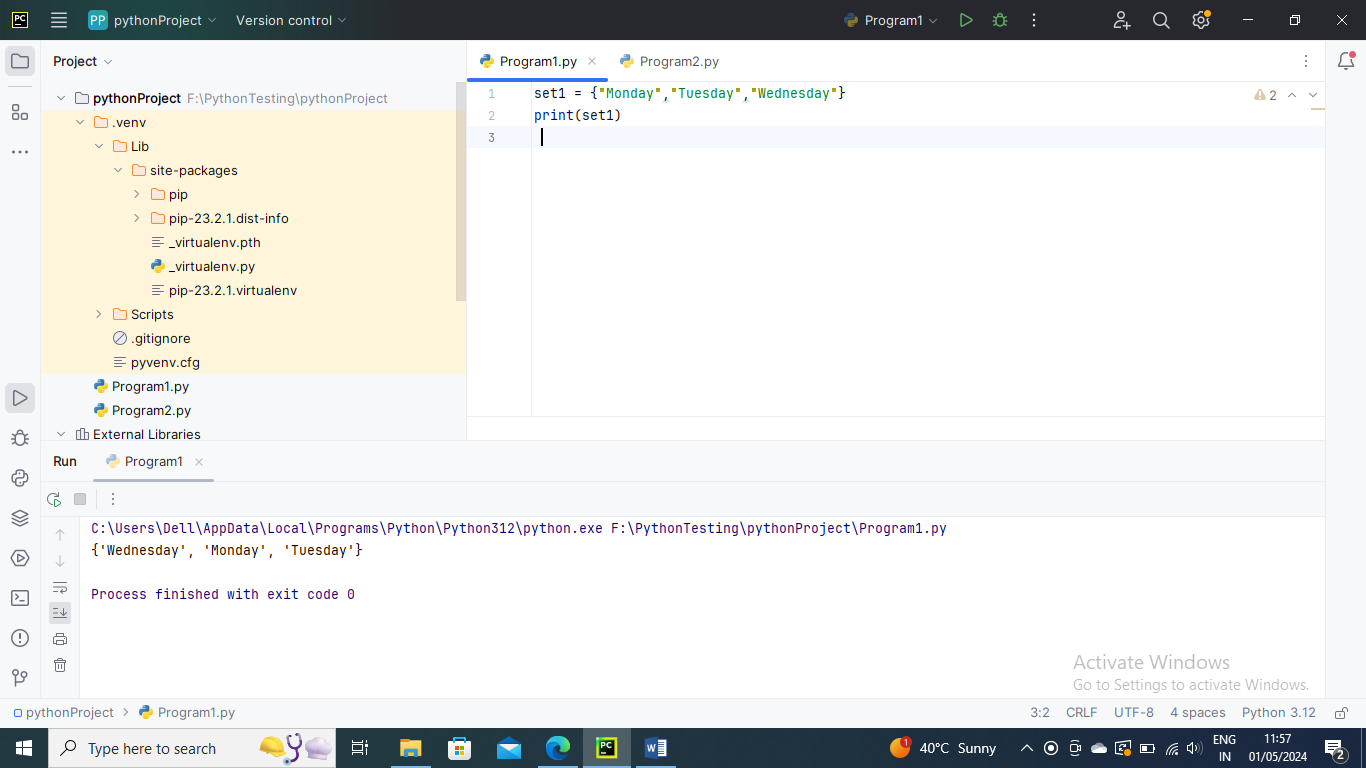
Dictionary –



Tuple –



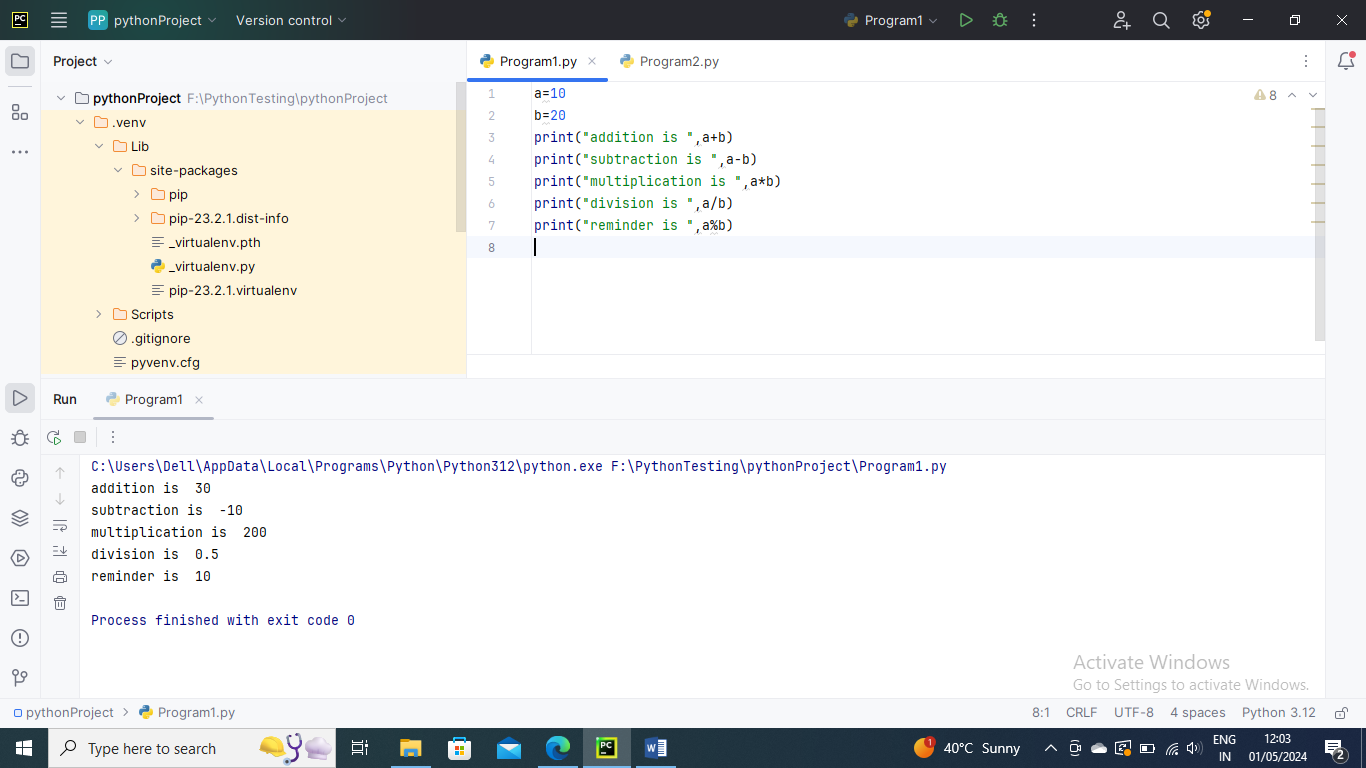
Set –



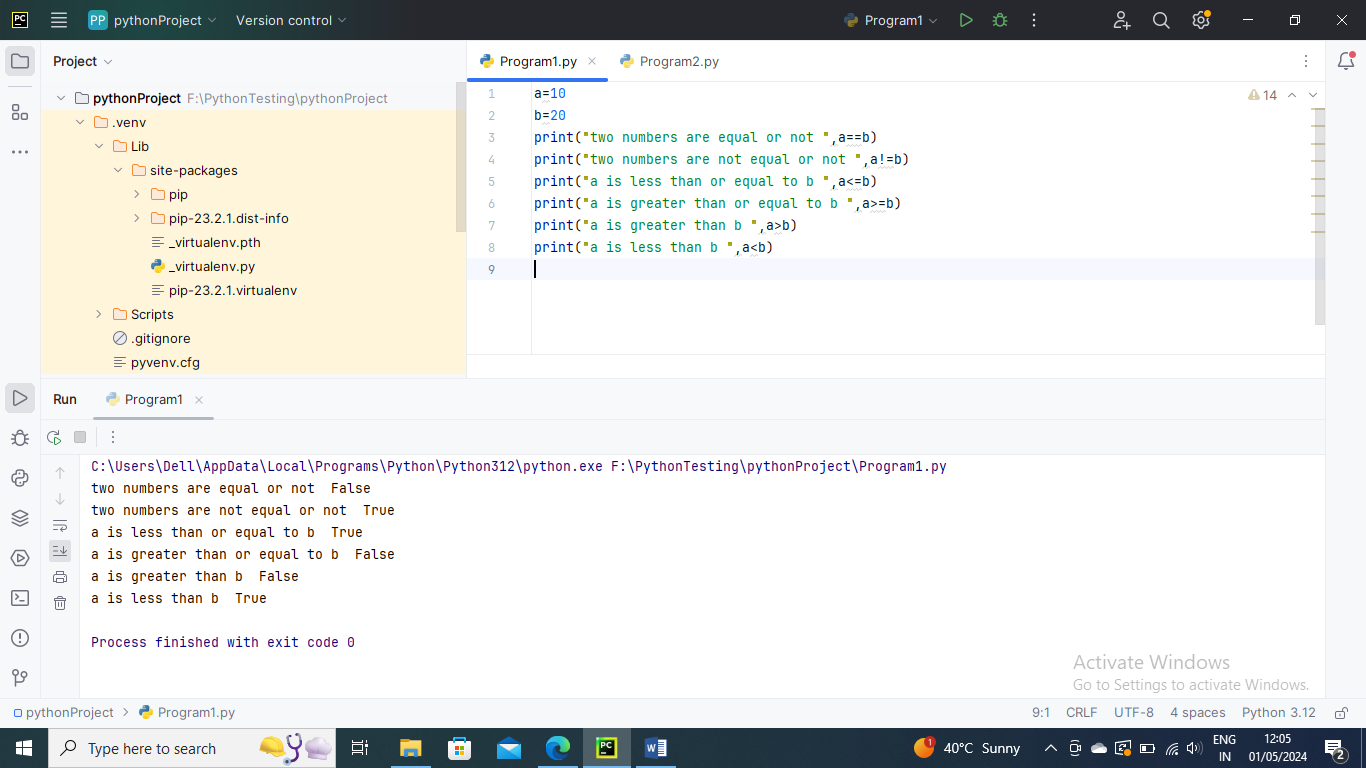
-----------------

Python Operators –

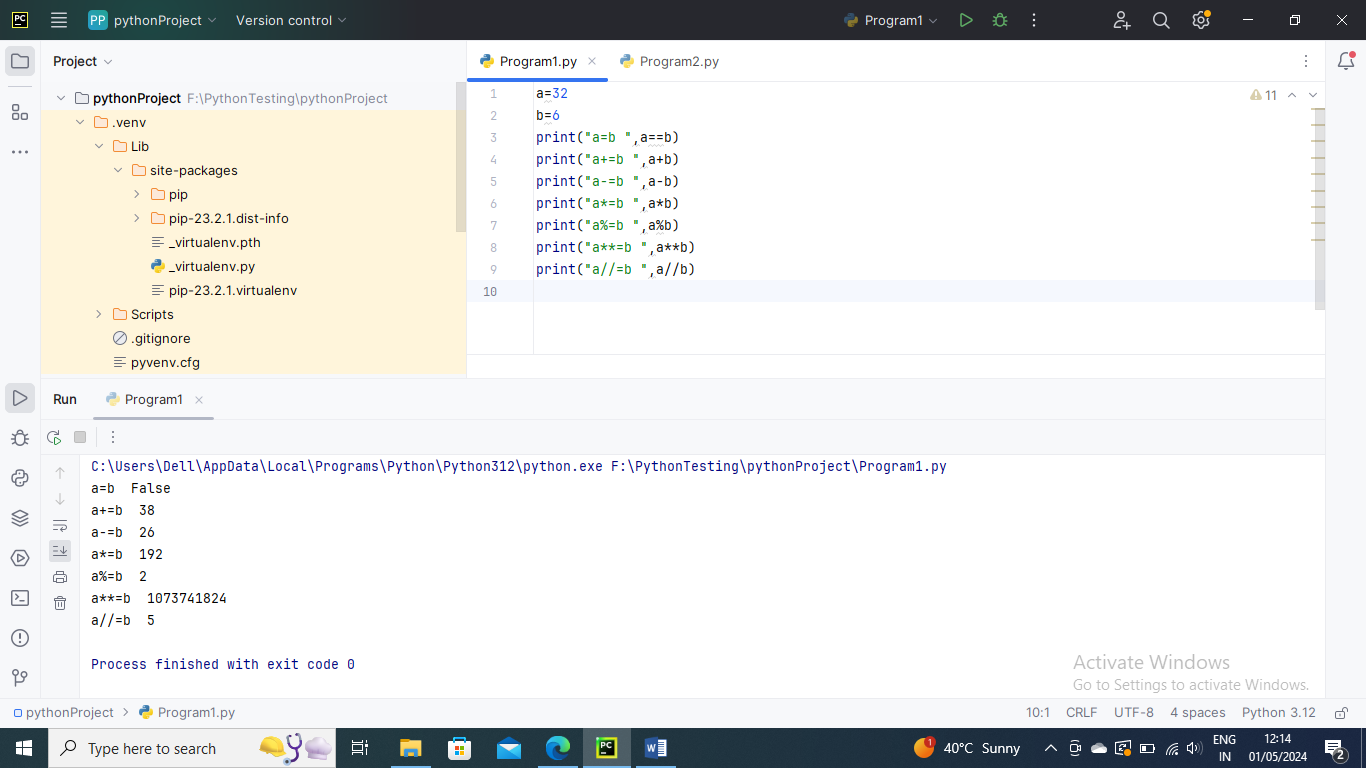
Arithematic Operators



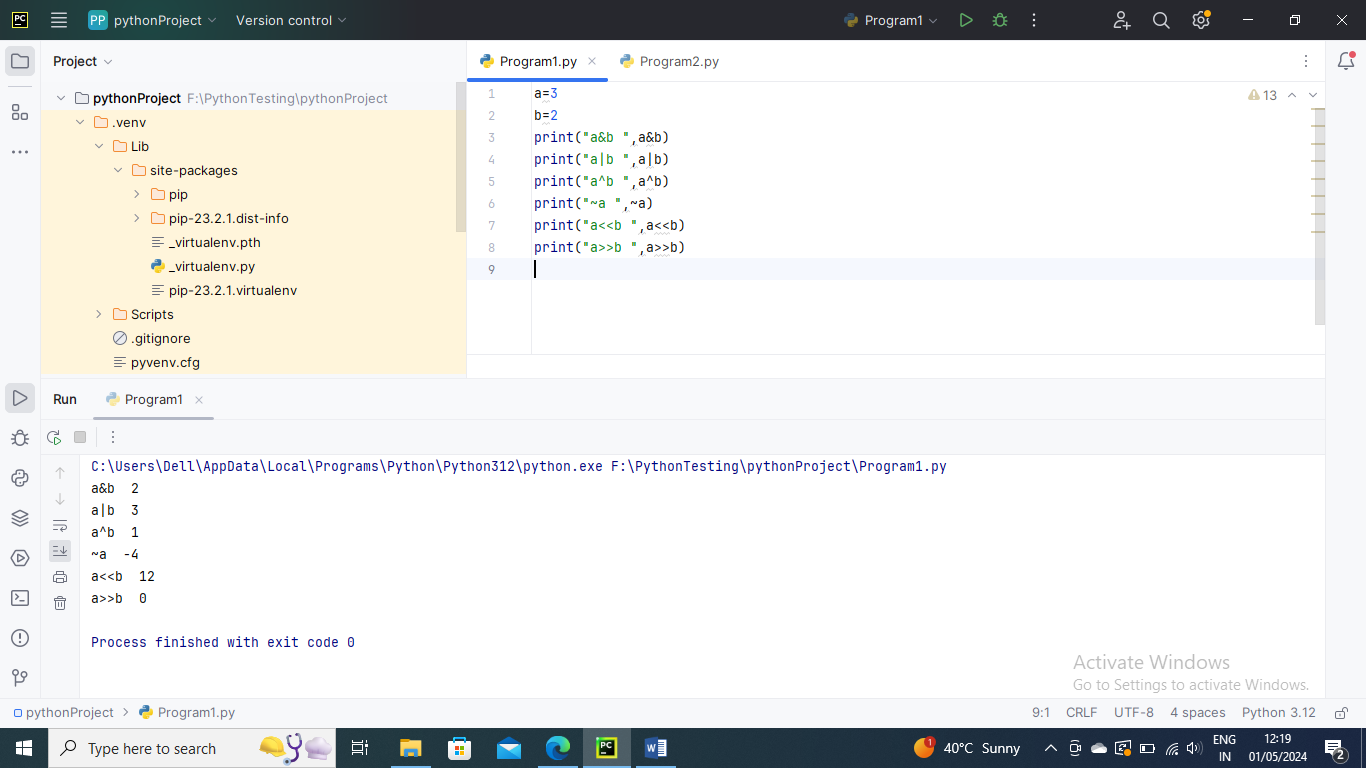
Comparison Operator



Assignment Operator



Bitwise Operator –

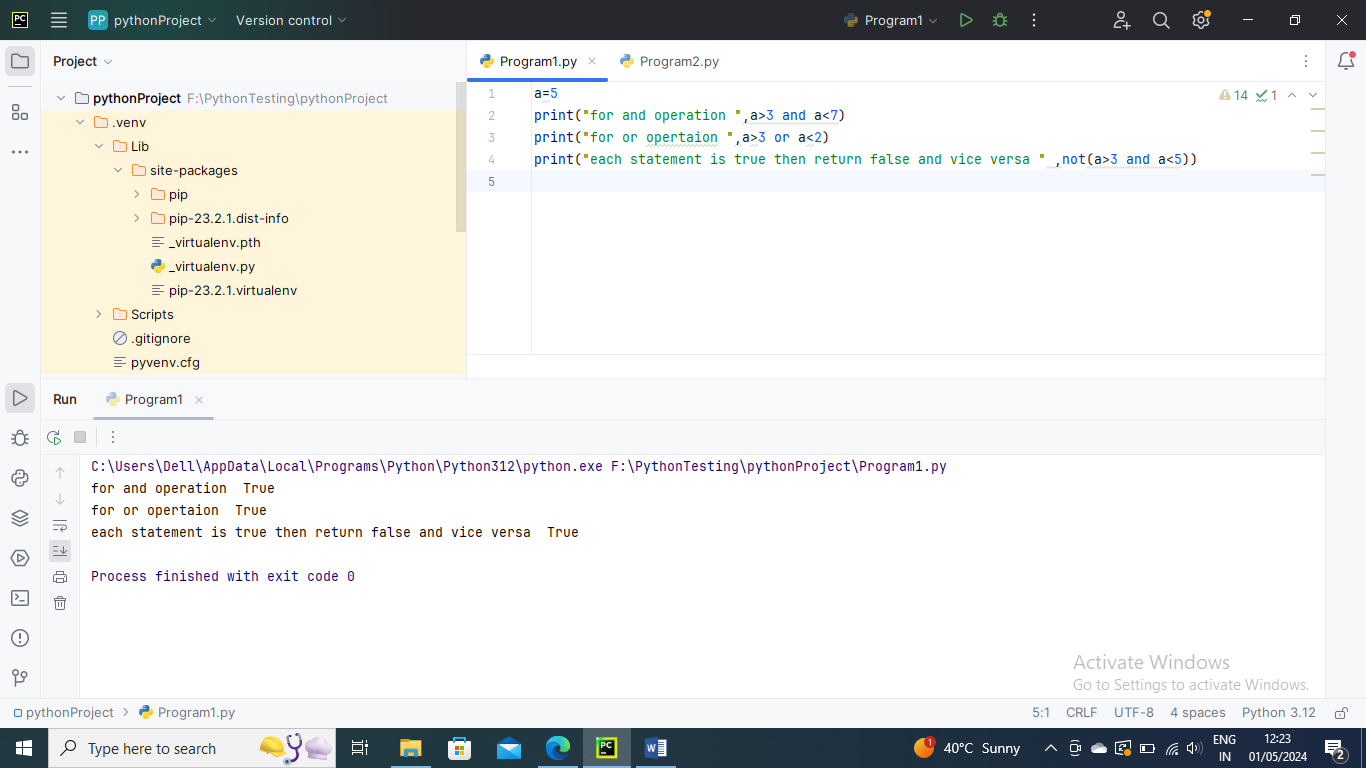


Logical Operaters

And – bothe should be true

Or – any one should be true

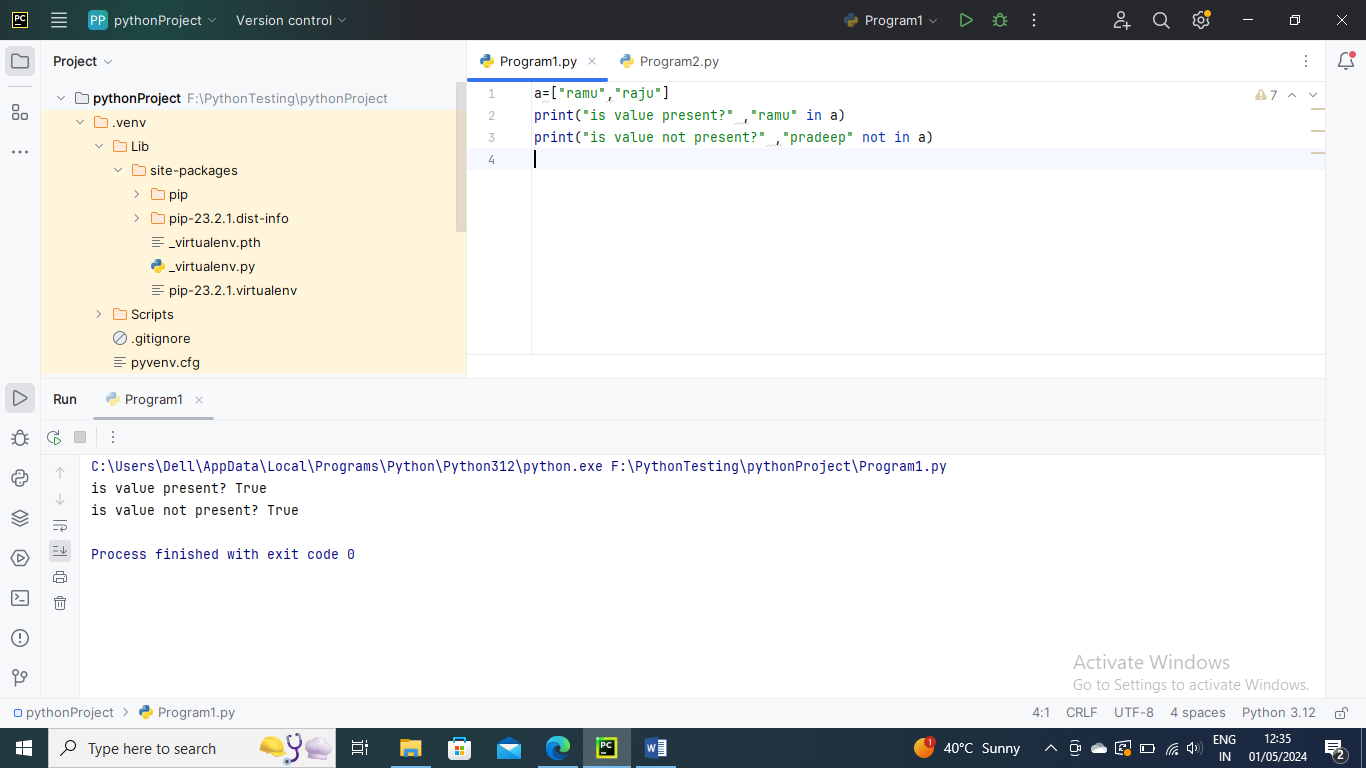
Not – If an expression a is true the, not(a) will be false and vice versa



Membership Operators

in – If the first operand cannot be found in the second operand, it is evaluated to be true (list, tuple or dictionary)

not in – If the first operand is not present in the second operand, the evaluation is true (list, tuple or dictionary)



Identity Operators

is – if the references on both sides point to same object, it is determined to be true

is not - if the references on both sides do not point at same object, it is determined to be true

